A Graph Based Approach for Naming Faces in News Photos

Derya Ozkan and Pınar Duygulu
Bilkent University, Department of Computer Engineering
06800, Ankara, Turkey
{deryao,duygulu}@cs.bilkent.edu.tr

Abstract

We propose a method to associate names and faces for querying people in large news photo collections. On the assumption that a person's face is likely to appear when his/her name is mentioned in the caption, first all the faces associated with the query name are selected. Among these faces, there could be many faces corresponding to the queried person in different conditions, poses and times, but there could also be other faces corresponding to other people in the caption or some non-face images due to the errors in the face detection method used. However, in most cases, the number of corresponding faces of the queried person will be large, and these faces will be more similar to each other than to others. In this study, we propose a graph based method to find the most similar subset among the set of possible faces associated with the query name, where the most similar subset is likely to correspond to the faces of the queried person. When the similarity of faces are represented in a graph structure, the set of most similar faces will be the densest component in the graph. We represent the similarity of faces using SIFT descriptors. The matching interest points on two faces are decided after the application of two constraints, namely the geometrical constraint and the unique match constraint. The average distance of the matching points are used to construct the similarity graph. The most similar set of faces is then found based on a greedy densest component algorithm. The experiments are performed on thousands of news photographs taken in real life conditions and, therefore, having a large variety of poses, illuminations and expressions.

1. Introduction

Effective and efficient retrieval, organization and analysis of large quantities of multi-modal data constitutes a big challenge. News photographs on the web are rich sources of information and accessing them is especially important. News mostly consist of stories about people; therefore, queries related to a specific person are desired. The usual way to retrieve information related to a person is to search using his/her name in the caption. However, such an approach is likely to yield incorrect results. In order to retrieve the correct images of a particular person, visual information must be incorporated and the face of the person needs to be recognized.

Face recognition is a long standing and well studied problem (see [4, 11] for recent surveys). However, most of the face recognition methods are evaluated only in controlled environments and for limited data sets. For larger and more realistic data sets, face recognition is difficult and error-prone due to large variations in pose, illumination and facial expression.

Recently, it has been shown that use of multi-modality allows better retrieval and analysis of the data and makes automatic labeling of images and objects possible [1]. When text information is provided together with the visual appearance, the face recognition problem can also be simplified and it can be transformed into the problem of finding associations between names and faces [8, 2]. As in [10], the combination of text and face information can also allow better retrieval performances without requiring recognition.

In this study, we propose a method to retrieve the correct faces of a queried person using both text and visual appearances. The data set used, namely the news photographs collected by Berg *et al.* [2], is quite different from most of the existing data sets (see Figure 1). It consists of large number of photographs with associated captions collected from Yahoo! News on the Web. Photographs are taken in real life conditions rather than in restricted and controlled environments. Therefore, they represent a large variety of poses, illuminations and expressions. They are taken both indoors and outdoors. The large variety of environmental conditions, occlusions, clothing, and ages make the data set even more difficult to be recognized.

Figure 1. Example faces from news photographs [2].

Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)
0-7695-2597-0/06 $20.00 © 2006 IEEE

Our method is based on the idea that a person's face frequently appears around his/her name although there may be other faces corresponding to other people in the story, or some non-face images due to the errors of the face detection methods used, like those shown in Figure 2. The conditions or the poses can vary, but the different representations of the face of the same person tend to be more similar to each other than to the faces of others.

Figure 2. Sample faces that are associated with the name *President George W. Bush*.

Based on these assumptions, we propose a graph based method for finding the group of most similar faces associated with a given name. Our method is based on representing the similarities between the faces associated with a given name in a graph and then finding the densest component which corresponds to the group of most similar faces. We first use the queried person's name to limit the search space. We assume that in this subspace faces of the queried person appear more than faces of any other person. Once a similarity measure is assigned between each pair of faces in the search space, the search space can be represented as a weighted graph, in which the nodes are faces and the assigned dissimilarities are edge weights. Usually, the nodes (faces) of the queried person will be similar to each other and different from other nodes in the graph. Since the queried person is the one whose face appears most frequently in the search space, the densest component of the full graph – the set of highly connected nodes in the graph – will correspond to the face of the queried person.

We represent the similarities between faces using interest points. Methods that use salient features for recognizing faces are reported in the literature [6]. In this study, we use Lowe's SIFT descriptors, which have been shown to be successful in recognizing objects [5, 7] and faces [9]. The interest points having the minimum distance are assumed to be the initial matching points. Two constraints, the geometrical constraint and the unique match constraint, are applied to select the best matching points.

The proposed method is not a solution to the general face recognition problem. Rather, it is a method to increase the retrieval performance of the person queries in the large data sets where names and faces appear together and where traditional face recognition systems cannot be used. It does not require a training step for a specific person and therefore, there is no limit on the number of people queried.

In the following, we first explain the data set and how names and faces are integrated. Then, we describe our method of making a graph based on the similarities of the faces and we explain the algorithm we use for finding the densest component. From this, we extract the group of faces corresponding to the person that we are searching for. Finally, we give the results of our experiments on news photographs [2].

2. Integrating Names and Faces

The data set consists of news photographs with captions. There can be more than one face in the photograph and more than one name in the caption; therefore, it is not known which face goes with which name. The first step in our method is to integrate the face and the name information. We use the name information mainly to limit the search space, since a person is likely to appear in a photograph when his/her name is mentioned in the caption. With this assumption, we reduce the face set for a queried person by only choosing the photographs that include the name of that person in the associated caption.

A person's name can appear in different forms. For example, the names *George W. Bush, President Bush, U.S. President*, and *President George Bush*, all correspond to the same person. We merged the set of different names used for the same person to find all faces associated with all different names of the same person.

Integrating names and faces produces better retrieval performances compared to solely text-based methods since non-face images are eliminated. However, the resulting set may still contain many false faces as mentioned previously. In the following sections, we explain our strategy to increase retrieval performance by finding the correct faces by using visual similarities.

3. Representing the Similarity of Faces

Our method is based on the fact that different instances of the face of a particular person are more similar to each other than to others. In this study, we represent the faces with the interest points extracted from the images using the SIFT operator [5]. For each pair of faces, the interest points on the first face are compared with the interest points on the second face and the points having the least Euclidean distance are assumed to be the correct matches. However, among these there can be many false matches as well (Figure 3). In order to eliminate the false matches, we apply two constraints: the geometrical constraint and the unique match constraint.

3.1. Geometrical Constraints

We expect that matching points will be found around similar positions on the face. For example, the left eye usually resides around the middle-left of a face, even in different poses. This assumption presumes that the matching pair of points will be in close proximity when the normalized

coordinates (the relative position of the points on the faces) are considered.

To eliminate false matches which are distant from each other, we apply a geometrical constraint. For this purpose, we randomly selected a set of images of 10 people (5 faces for each person). Then, we manually assigned true and false matches for each comparison and used them as training samples to be run on a quadratic Bayes normal classifier to classify a matched point as true or false according to its geometrical distance. The geometrical distance corresponding to the i^{th} assignment refers to $\sqrt{X^2+Y^2}$ where

$$X = \frac{locX\,(i)}{sizeX\,(image1)} - \frac{locX\,(match\,(i))}{sizeX\,(image2)},$$

$$Y = \frac{locY\,(i)}{sizeY\,(image1)} - \frac{locY\,(match\,(i))}{sizeY\,(image2)},$$

and $locX$ and $locY$ hold X and Y coordinates of the feature points in the images, $sizeX$ and $sizeY$ hold X and Y sizes of the images and $match(i)$ corresponds to the matched keypoint in the second image of the i^{th} feature point in the first image.

In Figure 3, matches before and after the application of this geometrical constraint are shown for an example face pair. Most of the false matches are eliminated when the points that are far away from each other are removed.

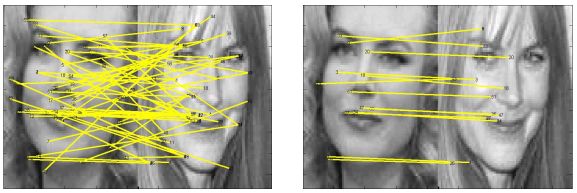

Figure 3. The first image on the left shows all the feature points and their matches based on the minimum distance. The second image on the right shows the matches that are assigned as true after the application of geometrical constraints.

3.2. Unique Match Constraints

After eliminating the points that do not satisfy the geometrical constraints, there can still be some false matches. Usually, the false matches are due to *multiple assignments* which exist when more than one point (e.g, A_1 and A_2) are assigned to a single point (e.g, B_1) in the other image, or to *one way assignments* which exist when a point A_1 is assigned to a point B_1 on the other image while the point B_1 is assigned to another point A_2 or not assigned to any point (Figure 4). These false matches can be eliminated with the application of another constraint, namely the unique match constraint, which guarantees that each assignment from an image A to another image B will have a corresponding assignment from image B to image A.

The false matches due to multiple assignments are eliminated by choosing the match with the minimum distance. The false matches due to one way assignments are eliminated by removing the links which do not have any corresponding assignment from the other side. An example showing the matches before and after applying the unique match constraints are given in Figure 5.

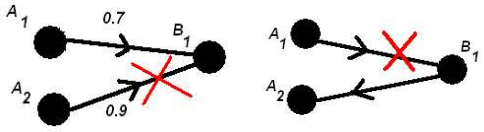

Figure 4. For a pair of faces A and B, let A_1 and A_2 be two points on A; and B_1 is a point on B with the arrows showing the matches and their direction. On the left is a *multiple assignment* where both points A_1 and A_2 on A match B_1 on B. In such a case, the match between A_2 and B_2 is eliminated. On the right is a *one way match* where B_1 is a match for A_1, whereas B_1 matches another point A_2 on A. The match of A_1 to B_1 is eliminated. The match of B_1 to A_2 remains the same if B_1 is also a match for A_2; otherwise it is eliminated.

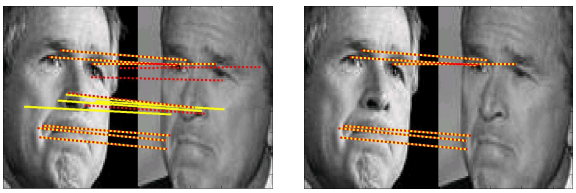

Figure 5. An example for unique match constraint. Matches from the left to the right image are shown by red, dashed lines, whereas matches from right to left are shown by yellow lines. The left image shows the matches assigned after applying geometrical constraints, but before applying the unique match constraints. The right image shows the remaining matches after applying the unique match constraints.

3.3. Constructing the Similarity Graph

After applying the constraints and assuming that the remaining matches are true matches, we define the distance between the two faces A and B as the average value of all matches.

$$dist(A,B) = \frac{\sum_{i=1}^{N} D(i)}{N},$$

where N is the number of true matches and $D(i)$ is the Euclidean distance between the SIFT descriptors of the two points for the i^{th} match.

A similarity graph for all faces in the search space is then constructed using these distances. We can represent the graph as a matrix as in Figure 6. The matrix is symmetric and the values on the diagonal are all zero. For a more clear visual representation, the distances for the faces corresponding to the person we are seeking are shown together. Clearly, these faces are more similar to each other

than to the others. Our goal is to find this subset which will correspond to the densest component in the graph structure.

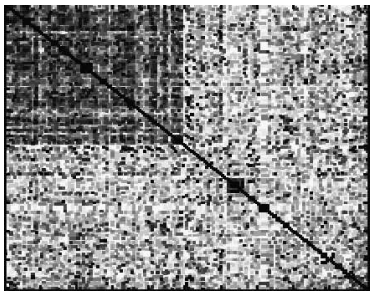

Figure 6. Dissimilarity matrix for 201 images in the search space for the name *Hans Blix*. In this search space, 98 of the images are true *Hans Blix* images, and the remaining 103 are not. For visualization, the 98 true *Blix* images are put on the top left of the matrix. Dark colors correspond to larger similarity values.

4. A Greedy Graph Algorithm to Find the Densest Component

Given the dissimilarity values between the face images, a graph structure is constructed, where faces represent nodes and the dissimilarities between the faces represent the edge weights (distance between two nodes). Our aim is to find the densest subgraph (component) since the nodes of the queried person will be close to each other and distant from all other nodes in the graph. In [3], the density of subset S of a graph G is defined as

$$f(S) = \frac{|E(S)|}{|S|},$$

where $E(S) = \{i, j \in E : i \in S, j \in S\}$ and E is the set of all edges in G. In other words, $E(S)$ is the set of edges induced by subset S. The subset S that maximizes $f(S)$ is defined as the densest component.

Our goal is to find the subgraph S with the largest average degree that is the subgraph with the maximum density. Initially, the algorithm presented in [3] starts with the entire graph G and sets $S = G$. Then, in each step, the vertex with the minimum degree is removed from S. The algorithm also computes the value of $f(S)$ for each step and continues until the set S is empty. Finally, the set S, that has maximum $f(S)$ value, is returned as the densest component of the graph.

However, the algorithm described above only works well for binary graphs. Thus, before applying it, we convert our original dissimilarity values into a binary form, in which 0 indicates no edge and 1 indicates an edge between two nodes. This conversion is carried out by applying a threshold on the distance between the nodes. This threshold also connotes what we define as near-by and/or remote. An example of such a conversion is given in Figure 7. In the example, assume that 0.65 is defined as our proximity threshold. In other words, if the distance between two nodes is less than or equal to 0.65 then these two nodes are near-by; therefore we put an edge between them. Otherwise, no edge is maintained between these nodes, since they are far away from each other.

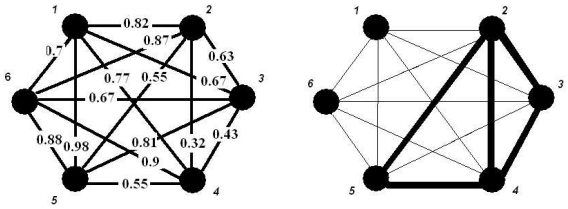

Figure 7. Example of converting a weighted graph to a binary graph. Nodes and their distances are given in the first image. The resulting graph after applying 0.65 as the proximity threshold is given in the second image. Bold edges are the edges that remain after conversion.

5. Experiments

The data set constructed by Berg *et al.* originally consists of about half a million captioned news images collected from Yahoo! News on the Web. After applying a face detection algorithm and processing the resulting faces, they were left with a total of 30,281 detected faces [2]. Each image in this set is associated with a set of names. A total of 13,292 different names are used for association. However more than half (9,609) of them are used only once or twice. Also, as we mentioned previously, a particular person may be called by different names. For example, the names used for *George W Bush* and their frequency are: *George W (1485); W. Bush (1462); George W. Bush (1454); President George W (1443); President Bush (905); U.S. President (722); President George Bush (44); President Bushs (2); President George W Bush (2); George W Bush (2).* We merge the set of different names used for the same person and then take the intersection to find faces associated with different names of the same person. Generally, the number of faces in the resulting set is less than the number of all names since a caption may include more than one instance of the referred name. For example, for *Bush* the number of faces is 2,849 while the total number of all referred names is 7,528. In the experiments, the top 23 people appearing with the highest frequencies (more than 200 times) are used. Figure 8 shows the total number of faces associated with the given name and the number of correct faces for the 23 people used in the experiments.

As the first step, the points having the minimum distance according to their SIFT descriptors are defined as the matching points. These points are further eliminated using the constraints. After this elimination process, 73% of all pos-

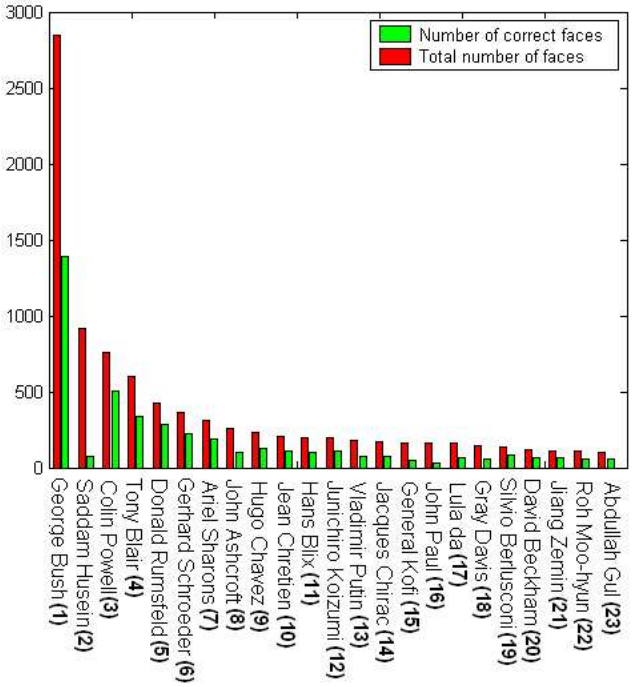

Figure 8. Names of 23 people are used in the experiments. The total number of faces associated with a name is represented by red bars and number of correct faces by green bars.

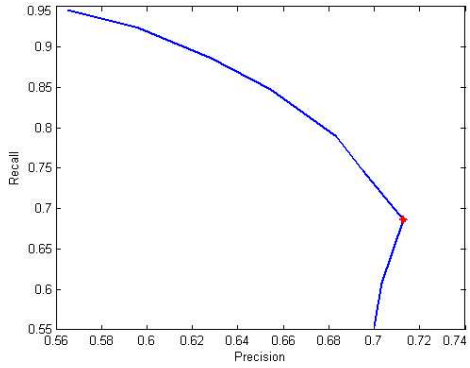

Figure 9. Recall-precision curve of 23 people in the test set. Precision and recall values change depending on the threshold. We used threshold values between 0.55 and 0.65 to show the effect. The threshold used in the rest of our experiments is marked with red.

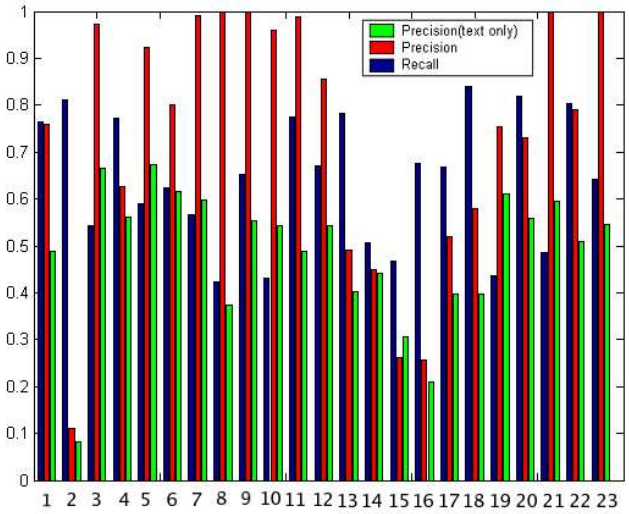

Figure 10. Recall and precision values for 23 people for threshold 0.575. Blue bars represent recall and red bars represent precision values that are achieved with the proposed method. Green bars are precision values for the baseline method, which does not use the visual information and retrieves the faces when name appears in the caption.

sible true matches are kept and we lose only 27% of true matches. Among these assignments, we achieved a correct matching rate of 72%.

Performance of the system changes depending on the threshold that we choose for converting the weighted graph – induced by the dissimilarity matrix – to a binary graph (see Figure 7 for this conversion). Average recall and precision values by varying the threshold between 0.55 and 0.65 is given in Figure 9. The threshold 0.575 is chosen to represent the recall and precision values for each person (Figure 10). For this threshold, the average precision value is 48% for the baseline method which assumes that all the faces appearing around the name is correct. With the proposed, method we achieved 68% recall and 71% precision values on the average. The method can achieve up to 84% recall- as for *Gray Davis*- and 100% precision - as for *John Ashcroft, Hugo Chavez, Jiang Zemin and Abdullah Gul*. We had initially assumed that, after associating names, true faces of the queried person appear more than any other person in the search space. However, when this is not the case, the algorithm gives bad retrieval results. For example, there is a total of 913 images associated with name *Saddam Hussein*, but only 74 of them are true *Saddam Hussein* images while 179 of them are *George Bush* images.

To show that our system works also on individuals appearing in a small number of captions, we performed experiments on 10 people appearing less than 35 times and obtained average recall and precision values 85% and 66%. As another experiment, we changed the number of instances of a face by removing some of the correct faces or by adding some incorrect faces. For 4 people having around 200 instances and similar number of true and false images (i) we removed 50 of true images of from each of their search space, (ii) we added 100 false images. Originally, average recall and precision values were 63% and 95%. We obtained 59% recall an 89% precision after (i), and 58% recall and 70% precision after (ii). Although the precision is somewhat affected, results are still acceptable.

For comparison, we perform another experiment where we use a supervised method by choosing positive and negative training examples of each person for a Bayes classifier. The recall and precision values by varying the percentage of training examples are given in Table 1. Although, the supervised method produces slightly better results than the proposed graph based method, it requires labeling and, therefore, it cannot be applied to a large number of faces.

Table 1. Supervised classification results based on the percentage of samples used in training. For example, for 25% positive and 25% negative training samples for each person, the system produces 66% recall and 80% precision on the average.

Percentage of Samples	25	50	75
Recall	0.66	0.73	0.75
Precision	0.80	0.82	0.80

The performance of our system is mainly based on computing the dissimilarity values since we compare each face with all other faces in the search space. For a search space with 200 pictures, the dissimilarity matrix is constructed in 9 minutes on a Pentium IV 3 GHz machine with 2 GB memory; and it takes less than 1 second to partition this graph.

6. Summary and Discussion

In this paper, we propose a graph based method for querying people in large news photograph collections with associated captions. Given dissimilarity (or similarity) measures between the face images in a data set, the problem is transformed into a graph problem in which we seek the densest component corresponding to the group of similar faces for a queried person. We use SIFT descriptors [5] to represent each face image and describe the dissimilarity values by using the average distances of the matching interest points. Then, we apply a greedy graph algorithm [3] to find the densest component.

For large realistic data sets, face recognition and retrieval is still a difficult and an error-prone problem due to large variations in pose, illumination and expressions. In this study, we have described a multi-modal approach for querying large numbers of people in such data sets. Compared to solely text-based methods, over 20% increase in precision is achieved. For individuals, up to 84% recall and 100% precision values can be obtained. The method does not require training for any specific person and thus it can be applied to any number of people. With this property, it is superior to any supervised method which requires labeling of large number of samples. The results achieved are also very close to supervised methods.

Before applying the greedy densest component algorithm, we convert the weighed graph consisting of dissimilarity values into a binary graph. However, this ignores some of the information. A method, which does not violate the weighted property of the graph, may yield better results. In this study, SIFT descriptors are used to represent the similarity of the faces. Other representations or similarity measures can also be used to construct the graph structure.

Acknowledgements

This work is supported by TÜBİTAK Career Grant 104E065 and Grant 104E077.

References

[1] K. Barnard, P. Duygulu, N. de Freitas, D. A. Forsyth, D. Blei, and M. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3, 2003. 1

[2] T. Berg, A. C. Berg, J. Edwards, and D. Forsyth. Who is in the picture. In *Neural Information Processing Systems(NIPS)*, 2004. 1, 2, 4

[3] M. Charikar. Greedy approximation algorithms for finding dense components in a graph. In *APPROX: Int. Workshop on Approximation Algorithms for Combinatorial Optimization*, London, UK, 2000. 4, 6

[4] R. Gross, S. Baker, I. Matthews, and T. Kanade. Face recognition across pose and illumination. In S. Z. Li and A. K. Jain, editors, *Handbook of Face Recognition*. Springer Verlag, 2004. 1

[5] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. Journal of Computer Vision*, 60(2), 2004. 2, 6

[6] B. S. Manjunath, R. Chellappa, and C. von der Malsburg. A feature based approach to face recognition. In *IEEE Conf. on Computer Vision and Pattern Recognition(CVPR)*, 1992. 2

[7] K. Mikolajczk and C. Schmid. A performance evaluation of local descriptors. In *IEEE Conf. on Computer Vision and Pattern Recognition(CVPR)*, 2003. 2

[8] S. Satoh and T. Kanade. Name-it: Association of face and name in video. In *IEEE Conf. on Computer Vision and Pattern Recognition(CVPR)*, 1997. 1

[9] J. Sivic, M. Everingham, and A. Zisserman. Person spotting: video shot retrieval for face sets. In *Int Conf. on Image and Video Retrieval(CIVR), Singapore*, 2005. 2

[10] J. Yang, M.-Y. Chen, and A. Hauptmann. Finding person x: Correlating names with visual appearances. In *Int. Conf. on Image and Video Retrieval(CIVR)*, Dublin City University Ireland, 2004. 1

[11] W. Zhao, R. Chellappa, P. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Computing Surveys*, 35(4), 2003. 1